



Learning Legible Motion from Human–Robot Interactions

Baptiste Busch, Jonathan Grizou, Manuel Lopes, Freek Stulp

► To cite this version:

Baptiste Busch, Jonathan Grizou, Manuel Lopes, Freek Stulp. Learning Legible Motion from Human–Robot Interactions. International Journal of Social Robotics, 2017, 211 (3-4), pp.517 - 530. 10.1007/s12369-017-0400-4 . hal-01629451

HAL Id: hal-01629451

<https://hal.science/hal-01629451>

Submitted on 6 Nov 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Learning Legible Motion from Human-Robot Interactions

Baptiste Busch · Jonathan Grizou · Manuel Lopes · Freek Stulp

the date of receipt and acceptance should be inserted later

Abstract In collaborative tasks, displaying legible behavior enables other members of the team to anticipate intentions and to thus coordinate their actions accordingly. Behavior is therefore considered to be *legible* when an observer is able to quickly and correctly infer the intention of the agent generating the behavior.

In previous work, legible robot behavior has been generated by using model-based methods to optimize task-specific models of legibility. In our work, we rather use model-free reinforcement learning with a generic, task-independent cost function. In the context of experiments involving a joint task between (thirty) human subjects and a humanoid robot, we show that: 1) legible behavior arises when rewarding the efficiency of joint task completion during human-robot interactions 2) behavior that has been optimized for one subject is also more legible for other subjects 3) the universal legibility of behavior is influenced by the choice of the policy representation.

Keywords Human-Robot Interaction (HRI) · legible motion · implicit coordination · reinforcement learning (RL)

This paper is an extended version of a previous work of Stulp et al. [21] and contains additional experimental results and more detailed discussions.

B. Busch, J. Grizou, M. Lopes, F. Stulp
Flowers Team, an Inria Bordeaux and ENSTA-Paristech joint-lab
E-mail: firstname.lastname@inria.fr

M. Lopes
INESC-ID, Instituto Superior Técnico, Universidade de Lisboa, Portugal

F. Stulp
German Aerospace Center (DLR), Institute of Robotics and Mechatronics, Wessling, Germany

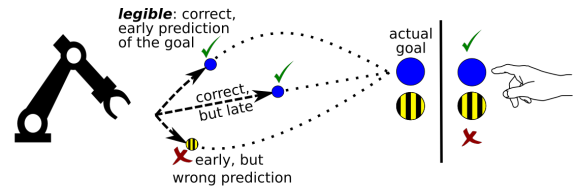


Fig. 1 Illustration of the button pressing experiment, where the robot reaches for and presses a button. The human subject predicts which button the robot will push, and is instructed to quickly press a button of the same color when sufficiently confident about this prediction. By rewarding the robot for fast and successful joint completion of the task, which indirectly rewards how quickly the human recognizes the robot’s intention and thus how quickly the human can start the complementary action, the robot learns to perform more legible motion. The three example trajectories illustrate the concept of legible behavior: it enables correct prediction of the intention early on in the trajectory.

1 Introduction

Humans exploit many non-verbal cues to efficiently coordinate their actions in joint tasks [16]. By monitoring the actions of others and inferring their intentions, a human can predict and preemptively initiate the appropriate complementary actions without the need for verbal communication [17, 16, 2]. Furthermore, it has been shown that humans unconsciously change their behavior, for instance the speed of task execution, to improve coordination [25].

The first contribution of this article is to show that robots may learn to adapt their behavior so that it becomes more legible, based only on observations of actual interactions with humans. We do so by proposing

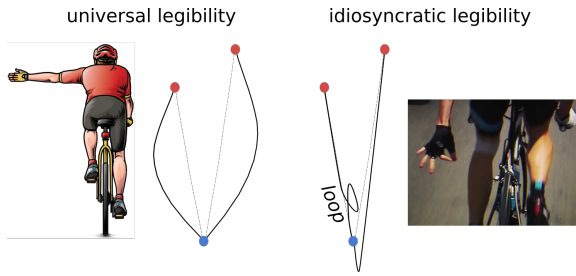


Fig. 2 Distinction between universal and idiosyncratic legibility. The left graph with trajectories has been adapted from [6].

a generic task-independent cost function, which is optimized with a model-free reinforcement learning algorithm.

Since our approach does not require a model, it is applicable to different tasks without modification. However, it does require a training phase to learn to generate legible behavior, and the resulting behavior generalizes to different tasks. A novel task thus requires learning a new behavior. In contrast, previous model-based methods [1, 3, 12, 14, 18, 19] are able to generate legible behavior on-the-fly, but require task-specific models of legibility. A novel task thus requires the design of a novel model by an expert.

Our approach is thus well suited for scenarios where not all tasks are known in advance, and where similar tasks are executed many times. In assembly lines where humans and cobots work together for instance, the resulting behavior is used thousands of times. The number of trials required to learn the behavior (<100) may thus well be worth the investment, and could also be performed on-the-job.

One question that arose whilst performing the experiments was whether robots learn to generate universally legible behavior, or rather idiosyncratic behavior that a human learns to interpret. The difference between the two is illustrated in Fig. 2. Even for cultures in which cycling is not widespread, an arm spread out to the left is likely to convey the intention that the cyclist will make a left turn. In contrast, the idiosyncratic signals exchanged between members of a cycling team during a race are not known to the general public (see Fig. 2, right), and only understood amongst other riders with whom these signals have been agreed upon beforehand.

Similarly, a robot may learn arbitrary but recognizable variations of the movement, such as the loop in Fig. 2 which the human may learn to be predictive of moving to the left. This idiosyncratic behavior will have to be relearned by other humans working with the same robot. In universally legible behavior on the

other hand, the intention is already understood during the first interaction(s).

The second contribution of this paper is to measure how well the legibility of behavior that has been learned from interactions with one subject transfer to other subjects, to determine whether the learned behaviors are universally or idiosyncratically legible.

The third contribution is to show how the representation of the robot’s controller influences whether universal or idiosyncratic legibility is achieved.

This article is structured as follows. After presenting related work in Section 2, we present four experiments¹ on the experimental setup illustrated in Fig. 1 and 3:

- Section 3: An experiment with 9 users, where the robot learns to be legible, using dynamical movement primitives as a policy representations.
- Section 4. As above, but using a viapoint policy, which is of much lower dimensionality.
- Section 5. Two experiments in which we study the transferability of legible robot behavior from one subject to another, with a total of 16 subjects. Second experiment gives some insight on the universal legibility of behaviors.

We conclude the article with Section 6.

2 Related work

In human-robot interaction, improving the human understanding of robot motion is a key feature. One way to achieve this can be to imitate the human motion in the same task context. The minimum jerk model [8] makes the assumption that human hand motion can be mathematically retrieved, by minimizing the jerk in Cartesian space, during a grasping task. On an industrial robot, however, trajectories generally follow a trapezoidal joint velocity profile [4]. Research has shown that predicting this type of motion is harder than a minimum jerk profile [9].

For specific tasks, it is possible to manually define motion that convey the desired intention. This can be made for different applications. For instance to facilitate handing over an object [1, 3, 12, 14, 18, 19], or to coordinate robot soccer players [20, 15]. The concept of legibility has also been studied in the context of safe navigation in the presence of humans [13]. Note that some researchers prefer to use the term “readability” rather than “legibility” [24].

Most similar to our purposes is the work of Dragan et al. [7]. They make a *general-purpose* definition of

¹ The experiment in Section 3 was previously reported [21]. Those in Section 4 and 5 are novel.

legibility: how probable is a goal, given a partially observed trajectory? Higher legibility implies that earlier in the trajectory it is already possible to distinguish the final goal. To note that legibility is different from predictability, clearly defined in that paper, predictability: what is the most probable trajectory, given knowledge of the goal? Although legibility and predictability are general ideas, they are implemented as cost functions that might not apply to all task contexts. It is a non-trivial task to adapt this cost function to novel task contexts, and especially to different (classes of) users. Robots are able to generate legible behavior by optimizing the legibility measure off-line through functional gradient optimization [6]. Alternatively, they can also generate deceptive behavior [5].

Following the work in [5,6], Zhao et al. [26] perform a human-robot experiment with Baxter torso-humanoid robot. For a large number of possible targets their results prove that a straight line pointing to the target is easier to predict than a trajectory obtained via maximizing the legibility criterion. Thus legibility seems to depend on the context of the task (e.g. number and position of possible targets).

We investigate legibility as an emergent adaptive property of interactions between people and robots. Rather than defining legibility as an explicit property to be optimized, we reward task efficiency. We apply model-free reinforcement learning methods, where the robot iteratively improves its legibility through trial-and-error interaction with a human. This approach has the advantage that no assumptions about the task or the human must be made, and the robot automatically adapts its legibility to the user preferences during the interaction. We evaluate our approach in several user studies with Baxter robot.

3 Experiment A: Learning Legible Motion

The hypothesis underlying this first experiment is that legibility of robot behavior needs not be defined and optimized explicitly, but that it arises automatically if joint task execution is penalized for not being efficient. To verify this hypothesis we have designed a joint human-robot task, in which the robot’s behavior is optimized – through model-free reinforcement learning – to minimize joint task execution duration. In this work, we use the term “joint task” to signify that both the robot and human must succeed at their subtask in order for the overall task to succeed, and that these subtasks depend on each other.

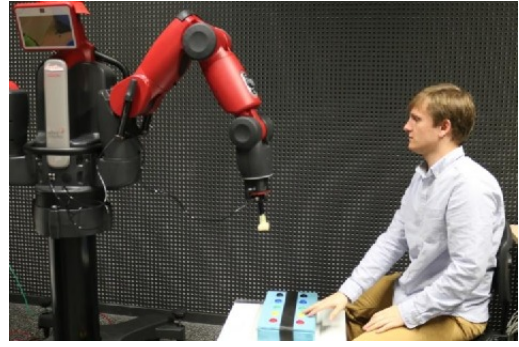


Fig. 3 Button pressing experiment set-up with the Baxter robot, human subject, and the two rows of buttons that they will press. The two possible targets corresponds to the “red” and “yellow” button on the box, the two buttons on the left side of the subject.

3.1 Methods

We now describe the experimental set-up, the policy representation that was used to generate the robot motion, the cost function that represents the task (fast joint task completion without errors), and the reinforcement learning algorithm used to iteratively optimize this cost function.

3.1.1 Experimental Set-up

In the joint human-robot task, depicted in Fig. 3, the robot reaches for and presses one of two buttons. Subjects are given two goals: **Efficiency**: press the same button as you think the robot will, as quickly as possible. **Robustness**: avoid making mistakes, i.e. pressing a different button from the one the robot will.

The 9 subjects for this experiment are administrative staff, PhD students in computer science, and undergrad students of cognitive science.

The protocol of an experiment is as follows. The experiment starts with a *habituation phase* of 32 trials where the robot performs always the same trajectory for the same button. This phase allows the subject to get used to the robotic motions, and practice the prediction and button pressing. It also allows to validate that the improvement in the subject’s prediction is not only due to them learning the robot’s motion. Further improvement after that habituation phase will then only be explained by the robot being more legible. Preliminary results indicate that 32 trials are sufficient for habituation[21].

After habituation, we start the *optimization phase* of 96 trials with the reinforcement learning algorithm presented in Section 3.1.3. The two policies that generate trajectories for the two different buttons are optimized in two independent processes.

3.1.2 Task Representation: Cost Function

The cost function that the robot optimizes during the 96 trials after the habituation phase consists of three components:

$$J = \underbrace{T_{\text{robot}} + T_{\text{subject}}}_{\text{Efficiency}} + \underbrace{\gamma \delta_{\text{buttons}}}_{\text{Robustness}} + \underbrace{\alpha |\ddot{\mathbf{q}}_{1\dots N, 1\dots T}|}_{\text{Energy}} \quad (1)$$

Efficiency: The time between the onset of the robot's movement (t_0) and the pushing of the button by the human (T_{subject}) and the robot (T_{robot}).

Robustness: Whether the subject pressed the correct button ($\delta_{\text{buttons}} = 0$) or not ($\delta_{\text{buttons}} = 1$). γ is an arbitrary high cost, it was set to 20 in this experiment, expressing that a failure is equivalent to a penalty of 20s in terms of efficiency.

Energy: The sum over the jerk, i.e. the third derivative of the joint positions ($\ddot{\mathbf{q}}_{t_i}$), at each time step i in the trajectory. The scaling factor α is chosen such that the cost of the jerk is about 1/20 of the total cost in the initial trajectories.

The joint task completion time depends mainly on how fast the human is able to predict the intention of the robot (proximate cause). But we use the total time because: 1) the ultimate motivation behind our research is to make human-robot interaction more efficient. 2) our set-up easily allows us to determine the button pressing times, but not the exact time at which the human predicts the robot's intention.

3.1.3 Optimization Algorithm: Direct Policy Search

The robot uses *direct policy search* to optimize the cost function in (1). Direct policy search is a form of reinforcement learning in which the search for the optimal policy is done directly in the space of the parameters θ of a parameterized policy π_θ , rather than using a value function. The specific algorithm we use is PI^{BB} (Policy Improvement through Black-Box optimization [22]). Since any model-free direct policy search algorithm could be used to implement this optimization (e.g. NES, CMA-ES or PoWER [23]), the details of PI^{BB} 's implementation have been deferred to Appendix A.

3.1.4 Policy Representation: Dynamical Movement Primitive

The parameterized policy representation π_θ used in this experiment is a dynamical movement primitive (DMP) [10]. DMPs combine a feedback controller (a spring-damper system with rest point g) with an open

loop controller (a function approximator f) to generate smooth goal-directed movements, see (2). The so-called phase system is 1 at the beginning of the movement and decays exponentially towards 0. The phase variable s is essentially an alternative 1-dimensional representation of time t .

$$\tau \ddot{x}_t = \underbrace{\alpha(\beta(g - x_t) - \dot{x}_t)}_{\text{feedback controller}} + \underbrace{s_t f(s_t)}_{\text{open loop controller}} \quad (2)$$

$$\tau \dot{s}_t = -\alpha_s s_t \quad \text{phase system} \quad (3)$$

When integrated over time, DMPs generate trajectories $[x_t \ \dot{x}_t \ \ddot{x}_t]$, which, for instance, are used as a desired joint angle or desired end-effector coordinate. In our experiments, 7 such systems are coupled to determine the 7 joint angles $x_t^{1:7}$ of the robot's arm over time.

The function approximator f takes the movement phase s as an input. In this paper, we use a radial basis function network with $B = 3$ Gaussian basis functions:

$$f(s_t) = \sum_{b=1}^B g_b(s_t) \theta_b \quad \text{RBFN} \quad (4)$$

$$g_b(s) = \exp\left(\frac{-(s - c_b)^2}{2\sigma_b^2}\right) \quad \text{Gaussian kernel} \quad (5)$$

The policy parameters θ thus correspond to the weights of the basis functions. Because there are 7 joints with 3 basis functions each, the dimensionality of θ is 21. During the optimization, variations in θ lead to variations in the trajectory towards the button.

DMPs are convenient for our experiments, as they ensure convergence towards the goal g (i.e. the location of the button), whilst allowing the trajectory towards this goal to be adapted by changing the parameters θ of the radial basis function network used inside the DMP (for instance to improve legibility). But our approach does not hinge on the use of DMPs as a policy representation, and we refer to [10] for details.

Please note that the same cost function, optimization algorithm and policy representation have been used for a very different task, i.e. the pick-and-place task described in [21]. Although the learned behavior for a task is specific to that task, our algorithms for learning these behaviors are not task-specific themselves.

3.2 Results

For illustration purposes, the top graph in Fig. 4 shows an example experiment for one subject, visualizing both the values of the time it takes the subject to push the button (T_{subject}) and whether the same buttons are

pushed. The transition from the habituation to the optimization phases is depicted as a dashed line.

The main results of Experiment A are summarized in the two lower graphs in Fig. 4, which highlight statistics at important transitions during learning: the start (trial 1 to 8), the last trial of the habituation phase (25-32), and the first (33-40), intermediate (81-88) and final (121-128) block of trials during the optimization phase. We also measure the trajectory completion at prediction time, i.e the relative amount of trajectory (timewise) observed by the subject when it presses the button. This measure is calculated using the formula $100(1 - \frac{T_{robot} - T_{subject}}{T_{robot}})$. The complete results are shown in the left column of Fig. ?? in Appendix B.

The box plots show the average value of $T_{subject}$ over all 9 subjects and over blocks of 8 trials. To allow comparison between subjects without introducing variance due to the natural overall differences in their button pressing time $T_{subject}$, we normalized the results of each subject by their intrinsic time after habituation, which is computed as the average of the last 8 values of $T_{subject}$ in the habituation phase. Thus, the normalized mean over the last 8 trials of the habituation phase is 100 for each subject by definition.

Finally, the bottom graph in Fig. 4 shows the number of prediction errors per block of 8, averaged over all subjects.

3.3 Discussion

The main conclusion we derive from Fig. 4 is that optimizing the robot’s motion leads to a substantial (20%) and significant ($p = 5e^{-8}$, Wilcoxon signed-rank test) drop in $T_{subject}$, i.e. the time it takes for the user to press the button, between the end of the habituation phase (25-32) and the end of the optimization (121-128). As T_{robot} is consistent throughout the experiment, this drop in $T_{subject}$ also induces a drop in the trajectory completion at prediction time (from 70% to 50%). This improved efficiency is not merely due to subjects simply guessing a button, because the number of mistakes does not increase over time ($p = 0.26$, Wilcoxon signed-rank test between end of habituation and end of optimization).

There is also a relatively small but significant ($p = 0.001$) decrease of the prediction time during the habituation phase, which indicates that the differences in the initial trajectories before optimization already enable the subject to predict the robot’s intention. The fact that $T_{subject}$ is further improved by 20% during the optimization shows that the optimized trajectories are more easily distinguishable, i.e. legible, than the initial trajectories.

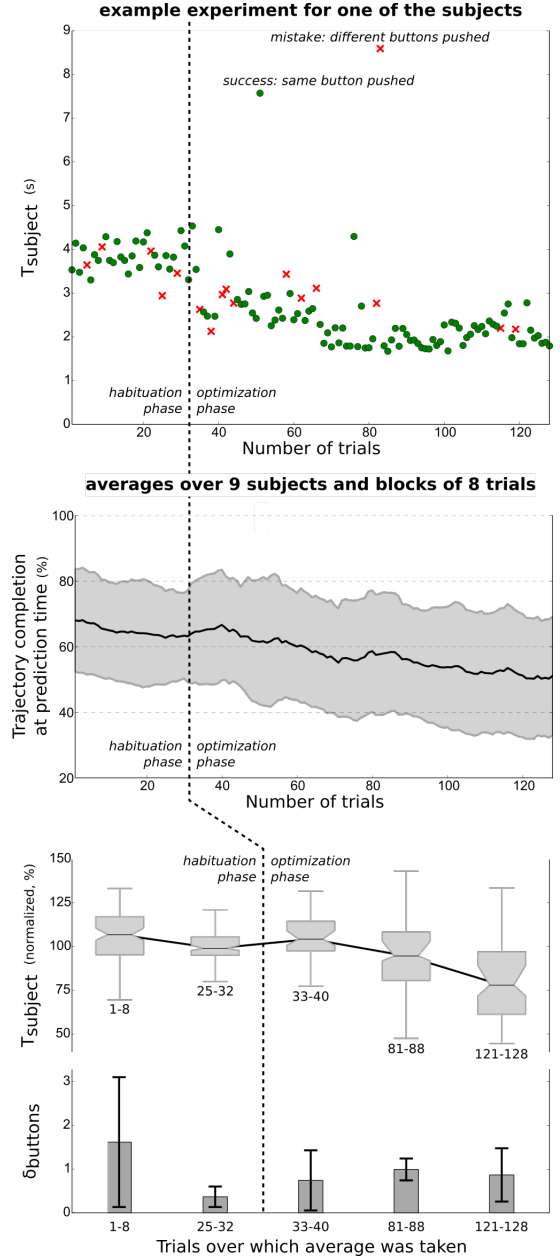


Fig. 4 Top: Example experiment for one subject, where $T_{subject}$ is plotted against the number of trials. Successful and failed trials are depicted as circles and crosses respectively. Middle: Average over all 9 subjects ($\mu \pm \sigma$) of the trajectory completion at prediction time, i.e, the relative amount of trajectory (timewise) observed by the subject when it presses the button. This value is calculated using the formula $100(1 - \frac{T_{robot} - T_{subject}}{T_{robot}})$. Bottom: Normalized $T_{subject}$ (see main text for normalization method), averaged over all 9 subjects and blocks of 8 trials; average number of failures, i.e. when different buttons were pushed, averaged over all 9 subjects and blocks of 8 trials. The lower two graphs show the values at certain key frames during learning; the complete results are presented in the left column of Fig. ?? in Appendix B.

After the habituation phase, subject’s performance got lowered (higher prediction time and higher number

of mispredictions). This effect arises from the variance of the parameters. As we do not model legibility, the robot can perform deceptive motions [5] while exploring the parameter space of the trajectories. This type of motion, which leads to higher cost under our cost function in 1, will slowly disappear after some iterations. Only the most legible trajectories remain, as confirmed by the drop in prediction time and the low misprediction rate.

In summary, the optimization algorithm effectively improves human-robot collaboration by producing motions that are easier to predict by the subject. By penalizing errors and the joint robot/human execution time, the robot learns policies that enable the human to distinguish the robot’s intentions earlier without more errors.

Although the answer to our initial question “Can a robot learn to generate legible motion from user interactions?” is positive, the resulting trajectories were nevertheless different from those observed in [6]. As an example, Fig. 5 plots two views of the robot’s trajectory. We clearly see a substantial upward movement at the beginning of the trajectory for button 1. This is certainly not universally legible behavior, but rather idiosyncratic behavior that the human subject learns to interpret as the motion that will eventually move towards button 1.

Further anecdotal evidence is that some subjects reported being able to infer the intention of the robot from differences in the *sound* produced by its motors. Differences in sound arise due to the different velocity profiles of the trajectories for the two buttons. This is clearly a very different type of legibility from that studied in [5,6,26]. Although this can be seen as another learned idiosyncrasy, it also suggests that legibility could be obtained by other means than only observing spatial variations of trajectories. This idea is also highlighted in Glasauer’s work [9] where they prove that minimum jerk velocity profiles are more legible than trapezoidal joint velocity one. Combining those elements could lead to even more legible trajectories.

For this reason, we designed a second experiment, discussed in the next section, which is aimed at avoiding such idiosyncratic behavior, and measuring the effects on learning legibility.

4 Experiment B: Learning Legible Motion with a Less Expressive Policy

The overall experimental set-up is the same as in Experiment A. Therefore, we only explain the differences, which are the policy representation, and a slightly modified cost function.

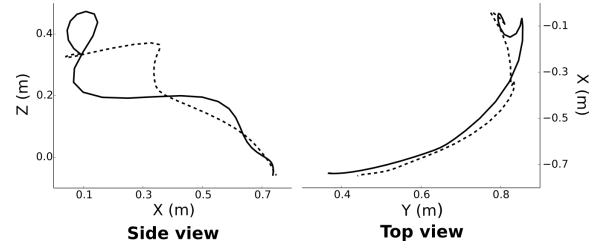


Fig. 5 Side and top view of generated trajectories after optimization for a single subject. Black/dashed: trajectory for button 1/2 respectively.

4.1 Methods

To avoid the idiosyncratic behavior observed with the DMPs, we designed a policy that allows for much less variations. The DMPs were defined in joint space (7 joints) with 3 basis functions that are varied per joint, leading to a policy that has $\theta=21$ parameters. To reduce this number, the second policy representation generates trajectories that pass through a viapoint, which itself is parameterized by only two parameters, as visualized in Fig. 6.

The trajectories are generated from a start point S (initial robot configuration) to an end point G (such that the button is pushed), which are fixed throughout the experiment. The height of the parabolic path is defined as a parameter h . The rotation around the x -axis, parallel to the ground, is defined as the parameter α . We represent this rotation seen from above. This policy constraints the generated trajectories for more smoothness. We expect them to resemble the ones obtain in Dragan’s work [7]. However we do not encode explicit informations about their legibility. Thus during the exploration of the parameter space some of the generated trajectories might be really deceptive. We call this policy the *viapoint policy*.

The cost function for the viapoint policy is the same as in Eq. 1, except that the penalty on the jerk is now in task space, not joint space. As before, the optimization of this cost function takes place within space of the policy parameters θ , which is now of dimensionality 2 (instead of 21 as with the DMP). We again use 9 subjects. To avoid any habituation effect from the first experiment we have chosen new participants.

4.2 Results

The main results of Experiment B are summarized in Fig. 7, which has the same format as Fig. 4. The complete results for this experiment are shown in the right column of Fig. ?? in Appendix B. Fig. ?? allows for a direct comparison of Experiment A and B.

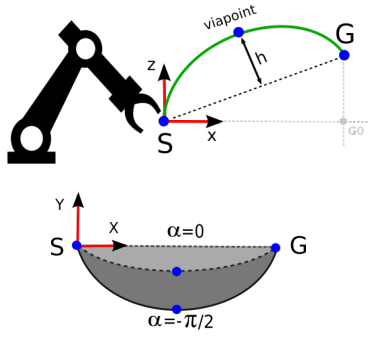


Fig. 6 Viapoint policy representation. Top: the trajectory is generated from the start S to the goal G (the location of the button), through a viapoint whose distance to the line $S-G$ is determined by the parameter h . The rotation around the x -axis is determined by α .

4.3 Discussion

We again observe a drop of the prediction time during optimization. Similarly to Experiment A this also creates a drop in the trajectory completion at prediction time (from 80% to 60%). The number of prediction errors increases during the optimization process before stabilizing at the end. The average number of errors is still sufficiently low, and not significantly different compared to the end of habituation ($p = 0.73$), to prove that the subjects are not simply guessing the next target. The decrease in prediction time during the habituation is significant ($p = 0.005$) as well as the decrease after the optimization ($p = 2.1e^{-5}$).

Qualitatively, these are thus the same results as in Experiment A. As for the DMPs, we represent in Fig. 8 two views of the trajectories. As expected, this policy produces smoother trajectories to the targets. In this case, the trajectories look like what we would expect from a legible behavior, i.e. an exaggeration on the right for the right target and on the opposite side for the left one.

The higher variance at the end of the optimization compared to Experiment A suggests not all subjects obtain such legible behaviors. In Fig. 9 we represent another example of optimized trajectories. The trajectories seen from above (top view) look rather deceptive.

One hypothesis is that by constraining the trajectories to resemble legible behavior we increase the number of local minima of the optimization. Consider that the global minima is achieved when the trajectories meet what we expect a legible motion to be. Because of the sampling in the parameter space that solution might not be found during the optimization. Moreover sub-

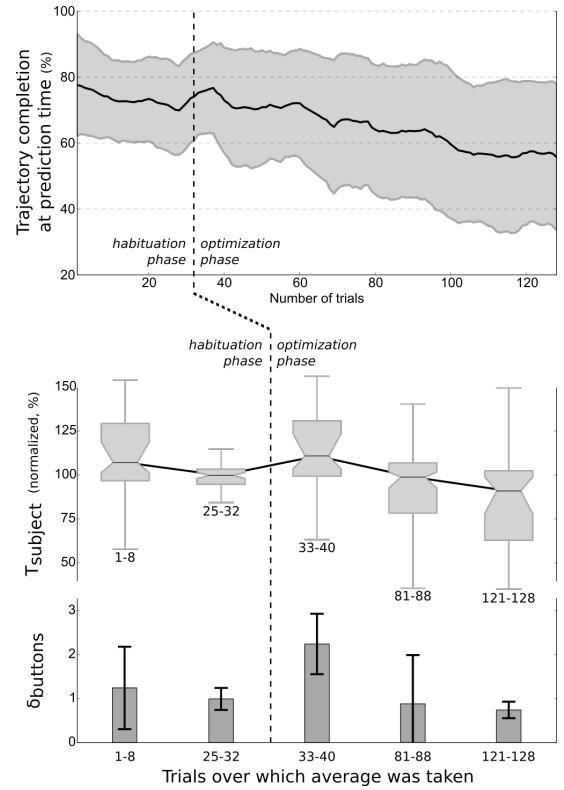


Fig. 7 Top: Average over all 9 subjects ($\mu \pm \sigma$) of the trajectory completion at prediction time, i.e. the relative amount of trajectory (timewise) observed by the subject when it presses the button. This value is calculated using the formula $100(1 - \frac{T_{robot} - T_{subject}}{T_{robot}})$. Bottom: Normalized $T_{subject}$, averaged over all 9 subjects and blocks of 8 trials; average number of failures, i.e. when different buttons were pushed, averaged over all 9 subjects and blocks of 8 trials. The lower two graphs show the values at certain key frames during learning; the complete results are presented in the right column of Fig. ?? in Appendix B.

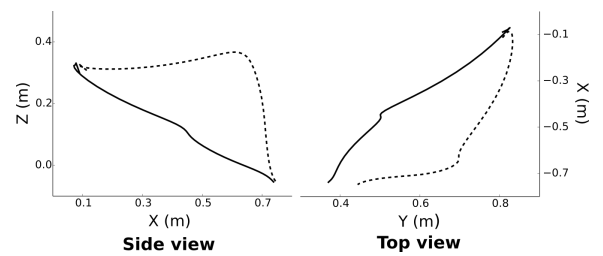


Fig. 8 Side and top view of generated trajectories after optimization for a single subject. Black/dashed: trajectory for button 1/2 respectively.

jects might learn a deceptive or idiosyncratic motion as they do with the DMP policy. Thus most of them decrease their prediction time at the end of the optimization. However the ones with the biggest drop obtain trajectories similar to those represented in Fig. 8.

The experiment in the next section will investigate how well trajectories generated by the two different op-

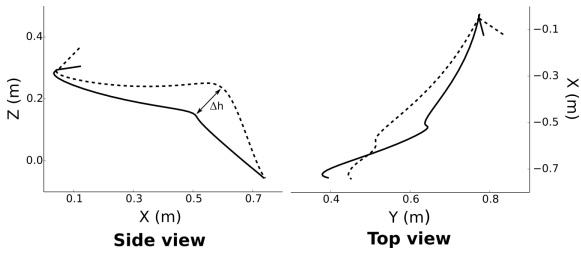


Fig. 9 Side and top view of generated trajectories after optimization for a single subject. Black/dashed: trajectory for button 1/2 respectively. The generated trajectory seems more deceptive when looking at the top view. Yet trajectories are distinguishable in term of height as represented by Δh in side view.

timized policies (DMP and viapoint) transfer to novel users.

5 Experiment C and D: Transferability of Legibility

Experiment A and B verify that robots are able to improve the legibility of their behavior from interactions with humans. We now present two experiments in which we investigate whether the adaptations that have been learned during interactions with one subject also improve the legibility for other subjects. The first experiment (Experiment C) uses the same protocol as A and B, but starts with trajectories that have been previously optimized. The second experiment (Experiment D) does not involve optimization, but rather presents several previously optimized trajectories in a random order. Experiment C is aimed at determining whether humans can learn to interpret the idiosyncratic motions of robots, whereas D aims at which type of trajectories enable humans to immediately recognize intentions, without the need to learn how to interpret them.

5.1 Methods

Experiment C Do subjects learn quicker when starting with policies that have been optimized previously with another subject? To analyze this, we ran the same experimental protocol with the habituation and optimization phase as described in Section 3.1, with 4 novel subjects each for both policy parameterizations (DMP and viapoint policy). In contrast to the optimizations described previously, the initial trajectories are now trajectories that have been previously optimized for other subjects. The initial trajectories were not chosen randomly but correspond to the most legible ones for each parameterization, i.e. the ones that lead to the biggest drop in term of prediction time.

| Run | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| Targets | R | R | B | R | B | B | B | R | B | R |
| Types | DMP_2 | S | S | DMP_1 | VP_1 | VP_2 | DMP_1 | VP_2 | DMP_2 | VP_1 |

Table 1 Illustration of one random sequence for experiment D. A complete run comprises a repetition of four such random sequences. This makes a total of 40 trials.

Experiment D The aim of this experiment is to determine if subjects can immediately recognize the intention of the robot from trajectories optimized for other subjects. Therefore, we use neither a habituation nor optimization phases for one particular trajectory, but rather present different previously optimized trajectories only a few times. A limited number of presentations is necessary, because the human may learn to interpret potential idiosyncrasies of the movements, which we want to avoid in this experiment.

For both buttons, five types of trajectories are presented:

- trajectories generated by two optimized DMP policies (from Experiment A) that lead to the largest reduction in T_{subject} . We refer to them as DMP_1 and DMP_2
- as above but with two viapoint policies (from Experiment B) noted VP_1 and VP_2
- straight line minimum-jerk trajectories (S) with end-effector pointing toward the button, as a baseline.

The order of the buttons (denoted **R** and **B**) and trajectory types is random within a sequence of 10 trials. The sequence is repeated 4 times which lead to a complete run comprising 40 trials. An example of a random sequence is presented in Table 1. The work of Zhao et al. [26] shows that straight line minimum-jerk trajectories, with end-effector pointing toward the target, are the most legible for a high number of possible targets. By comparing the DMPs and the viapoint based trajectories to this kind of straight lines, we hypothesize that for the two-target case scenario the other types of trajectories convey more informations and thus are more legible. For this experiment, 8 novel subjects were used.

5.2 Results

The results of Experiment C are plotted in Fig. 10. Whereas previous experiments showed smaller improvements during habituation (7% and 10% for DMP and viapoint respectively) and large improvements during optimization (a further 20% and 20%), we here see the inverse. The improvement during habituation is now 37% and 43% (both $p < 10^{-5}$), whereas during optimization they are small and not significant ($p = 0.47$).

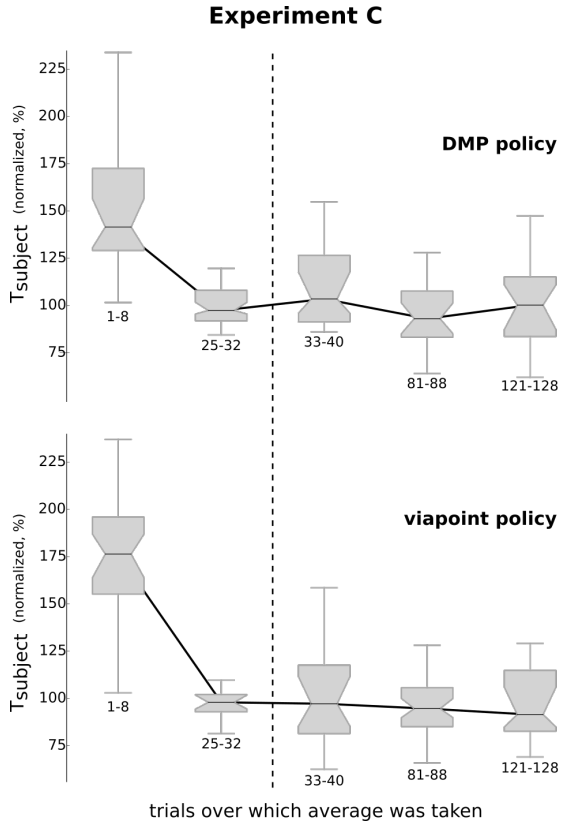


Fig. 10 Box plots for the normalized prediction times, when starting with previously optimized trajectories from the beginning, averaged over all subjects, and blocks of 8 trials. Top) DMP based trajectories. Bottom) viapoint trajectories.

and $p = 0.52$). The complete results of Experiment C are shown in Fig. ?? in Appendix C.

The results of Experiment D are summarized in Fig. 11. The top graph, depicts T_{subject} for all types of trajectories. Each bar represents the average over all users and buttons. Differences between buttons were not significant ($p > 0.33$, Wilcoxon signed-rank test), and thus pooled. Differences between the DMP and the two other type of trajectories are significant ($p < 0.03$, Welch's t -test). However the difference between the viapoint policy and the straight line is not significant ($p = 0.21$). The bottom graph depicts the same results for the number of errors. The differences between the viapoint policy and the two other type of trajectories is significant ($p < 0.03$, Welch's t -test). However there is no significant difference between the DMP and the straight lines ($p = 0.33$)

5.3 Discussion

The results in Fig. 10 suggest that subjects can quickly learn to recognize the intentions of the robot from trajectories optimized for another subject, for both the

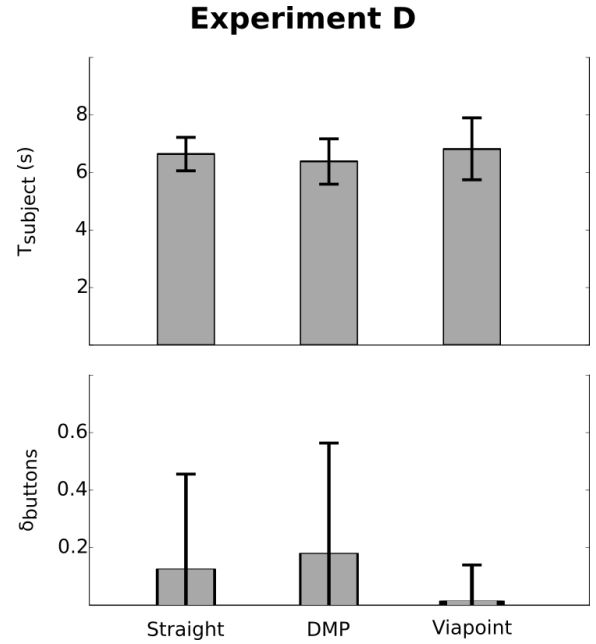


Fig. 11 Times (top graph) and prediction errors (bottom graph) for the three type of trajectories.

DMP and the viapoint policy. Because the improvement in T_{subject} during habituation is much more pronounced than during Experiment A and B, we deduce that these trajectories are indeed more legible.

From a comparison between T_{subject} of Experiment A and B and their equivalent in Experiment C we observe some interesting behaviors. First the difference in T_{subject} for the DMP on the first eight trials is significant ($p < 0.03$, Mann-Whitney U test) with T_{subject} being lower for Experiment A. We also note that the subject's predictions happen at 70% of the trajectory in Experiment A and 90% in Experiment C and that this difference is significant ($p < 0.03$, Mann-Whitney U test). Initial trajectories for Experiment A are close to straight line to the target (learned by demonstration). According to the definition of legibility this suggests that optimized trajectories might be less legible when shown to novel users without habituation. However humans adapt very quickly and by the end of the habituation the optimal time is reached and does not vary throughout the optimization. Moreover at the end of the habituation the prediction is performed at 50% of the robot trajectory when subject are shown optimized trajectories compared to 60% with the straight lines. We then deduce that optimized trajectories are more legible. This is however a contradiction with the fact that they started as less legible. As stated, the definition of legibility from Dragan et al. [7], cannot handle such contradictions because it does not account for the possibility of habituation. At the end of the optimiza-

tion phase the difference between Experiments A and C is not significant neither in term of time ($p = 0.42$) nor in term of trajectory completion at prediction time ($p = 0.08$).

For the viapoint policy the situation is the exact opposite. During the first eight trials the difference in T_{subject} is not significant ($p = 0.23$) neither is the difference in term of trajectory completion ($p = 0.41$). Thus optimized trajectories are at least as legible as straight lines without habituation. However at the end of the optimization there is a significant difference in term of time ($p < 0.03$) and therefore in term of trajectory completion trajectory with a drop of almost 10% (from 50% to 40%). The trajectories selected for Experiment C were the most legible one, i.e. the one that lead to the greatest drop in the subject's prediction time between habituation and optimization. This observation supports the hypothesis that the optimization of Experiment B have some local minima.

Are the viapoint trajectories more legible than the DMPs? For the DMP based trajectories, when looking at trials 8 to 16, the difference, in term of prediction time, between the straight lines of the habituation phase of Experiment A and the already optimized trajectories of Experiment C are not significant ($p < 0.03$). This means that after 8 trials of habituation subjects were able to perform similarly to those who observed straight lines to the target. But by the trials 16 to 24 they perform significantly better. For the viapoint policy it is sufficient to wait for the trials 8 to 16 to see a significant improvement in the prediction time. Thus we can conclude that the viapoint policy requires less habituation trials to perform better than the two other type of trajectories.

Between DMP and viapoint policies we note, at the end of the optimization, a difference in term of trajectory completion (50% with the DMP trajectories versus 40% with the viapoint ones). However this difference can be explained by the fact that T_{robot} is slightly different between the two policies. In fact, in term only of prediction time, both DMP and viapoint policies perform similarly (they both converge to 3.5s). Therefore, a direct comparison between them in term of prediction time might not be suitable as the subject's prediction time depends also on the speed of the movement of the robot.

The results in Fig. 11 are in accordance with the observations made in Experiment C. In term of prediction time all trajectories perform similarly. We recall that T_{robot} differs between the DMP and the viapoint policies. Thus comparing them only on time might be biased. However there is no ambiguity when looking at errors. The number of errors for the DMP policy is sim-

ilar to that of the straight trajectory, but the number of errors for the viapoint policy is far lower. This means that subjects are able to recognize the intention of the robot from the viapoint policy much more robustly than from the two other policies. Because subjects are able to do so immediately *without* habituation or previous training, this indicates that the viapoint policy is more legible than the two other policies.

From those results we conclude that reusing optimized trajectory on novel subjects allows for a faster learning of the robot's sense of legibility. Even with DMP based trajectories, where the robot's motion can be considered as idiosyncratic, subject were able to recognize faster the robot's intention. Moreover only the habituation phase is sufficient to reach the performances of the initial subjects for whom trajectories have been optimized. After habituation, no further improvement is achieved. The legibility of previously optimized trajectories could not be further increased by further optimization with another user. Another conclusion is that the viapoint policy is significantly more legible than the two other type of trajectories as it requires less habituation and leads to a lower error rate when presented without habituation.

6 Conclusion

In this article we studied how legibility can be obtained in a model-free approach. As any particular task will require different properties of motion, we want to achieve such results without any task-specific model of legibility. To such end we take an approach where we define a task-independent cost function that rewards efficiency (joint execution time), robustness (task errors), and energy (jerk). These measures can be readily defined for any task. To optimize such cost function through experiment we rely on a model-free optimization algorithm, PI^{BB} , to efficiently optimize this cost function through trial-and-error interaction of the robot with the human.

In several human-robot experiments, we showed that indeed, for different types of motions, robots are able to improve their behavior allowing humans to better read the robots' intentions early and robustly. Our results show that people, even after being habituated to robotic motions, can still substantially improve their prediction times if the robot optimizes its motions.

A second conclusion is that, when optimizing with policies that have a high-dimensional parameter vector (which leads to a lot of variance in the types of motions it can generate, such as with the DMP), it is most likely that idiosyncratic behavior arises. Novel subjects can infer the intention of the robot from its behavior, but this requires an extended phase of interaction with the

robot. These interactions are necessary for the novel subject to get to know the specific idiosyncrasies the robot has learned with the previous subject.

Furthermore, the robot is still able to learn legible behavior, even if we actively suppress idiosyncratic behavior by allowing only stereotypical curved minimum jerk movements. Already during first interactions, novel subjects are able to read such behavior more efficiently and robustly than when using the DMP policy. This indicates that this behavior is immediately and more generally legible.

Are the generated viapoint trajectories *universally* legible, i.e. across different robots or human cultures? Without any habituation, in term only of prediction time, they perform similarly to straight lines to the target. Although prediction time is a good indicator of legibility, there might be other factors that explain its variation. When working with real humans we also have to consider that some psychological effects can interfere with our expectations. For example, at the beginning of the task some subjects might wait for more confidence instead of trying to guess and potentially making mistakes. Moreover in all our experiment our subject's share similar background and culture. Would the generated behavior be still legible for people from other cultural background?

In general, we expect that the transition from idiosyncratic to universally legible behavior may not always be that well defined.

Acknowledgements This work is partially supported by INRIA, Conseil Régional d'Aquitaine, the ERC grant EXPLORERS 24007, and the 3rdHand European Project (funded under 7th FWP), Grant agreement no.: 610878.

References

1. Rachid Alami, Aurélie Clodic, Vincent Montreuil, and Emrah Akin Sisbot. Toward human-aware robot task planning. *AAAI Spring Symp. To boldly go where no human-robot team has gone before*, pages 39–46, 2006.
2. Cristina Becchio, Valeria Manera, Luisa Sartori, Andrea Cavallo, and Umberto Castiello. Grasping intentions: from thought experiments to empirical evidence. *Frontiers in human neuroscience*, 6, 2012.
3. M. Cakmak, S.S. Srinivasa, M.K. Lee, S. Kiesler, and J. Forlizzi. Using spatial and temporal contrast for fluent robot-human hand-overs. In *Proceedings of the 6th international conference on Human-robot interaction*, pages 489–496. ACM, 2011.
4. J. J. Craig. *Introduction to Robotics: Mechanics and Control*. Prentice Hall, 3rd edition, 2005.
5. Anca Dragan, Rachel Holladay, and Siddhartha Srinivasa. An analysis of deceptive robot motion. In *Robotics: Science and Systems*, July 2014.
6. Anca Dragan and Siddhartha Srinivasa. Generating legible motion. In *Robotics: Science and Systems*, June 2013.
7. Anca D Dragan, Kenton CT Lee, and Siddhartha S Srinivasa. Legibility and predictability of robot motion. In *Human-Robot Interaction (HRI), 2013 8th ACM/IEEE International Conference on*, pages 301–308. IEEE, 2013.
8. T. Flash and N. Hogan. The coordination of arm movements: an experimentally confirmed mathematical model. *The journal of Neuroscience*, 5(7):1688–1703, 1985.
9. S. Glasauer, M. Huber, P. Basili, A. Knoll, and T. Brandt. Interacting in time and space: Investigating human-human and human-robot joint action. In *IEEE International Workshop on Robot and Human Interactive Communication*, 2010.
10. A. Ijspeert, J. Nakanishi, P. Pastor, H. Hoffmann, and S. Schaal. Dynamical Movement Primitives: Learning attractor models for motor behaviors. *Neural Computation*, 25(2):328–373, 2013.
11. J. Kober and J. Peters. Policy search for motor primitives in robotics. *Machine Learning*, 84(1):171–203, 2011.
12. M.K. Lee, J. Forlizzi, S. Kiesler, M. Cakmak, and S. Srinivasa. Predictability or adaptivity?: designing robot hand-offs modeled from trained dogs and people. In *Proceedings of the 6th international conference on Human-robot interaction*, pages 179–180. ACM, 2011.
13. Christina Lichtenthäler, Tamara Lorenzy, and Alexandra Kirsch. Influence of legibility on perceived safety in a virtual human-robot path crossing task. In *Proceedings of the IEEE International Workshop on Robot and Human Interactive Communication*, pages 676–681, 2012.
14. Jim Mainprice, Emrah Akin Sisbot, Thierry Siméon, and Rachid Alami. Planning safe and legible hand-over motions for human-robot interaction. *IARP Workshop on Technical*, 2010.
15. Enrico Pagello, Antonio D'Angelo, Federico Montesello, Francesco Garelli, and Carlo Ferrari. Cooperative behaviors in multi-robot systems through implicit communication. *Robotics and Autonomous Systems*, 29(1):65–77, 1999.
16. Luisa Sartori, Cristina Becchio, and Umberto Castiello. Cues to intention: the role of movement information. *Cognition*, 119(2):242–252, 2011.
17. N. Sebanz, H. Bekkering, and G. Knoblich. Joint action: bodies and minds moving together. *Trends in Cognitive Sciences*, 10(2):70–76, 2006.
18. Kyle Strabala, Min Kyung Lee, Anca Dragan, Jodi Forlizzi, and Siddhartha S Srinivasa. Learning the communication of intent prior to physical collaboration. In *RO-MAN, 2012 IEEE*, pages 968–973. IEEE, 2012.
19. Kyle Wayne Strabala, Min Kyung Lee, Anca Diana Dragan, Jodi Lee Forlizzi, Siddhartha Srinivasa, Maya Cakmak, and Vincenzo Micelli. Towards seamless human-robot handovers. *Journal of Human-Robot Interaction*, 2(1):112–132, 2013.
20. F. Stulp, M. Isik, and M. Beetz. Implicit coordination in robotic teams using learned prediction models. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 1330–1335. IEEE, 2006.
21. Freek Stulp, Jonathan Grizou, Baptiste Busch, and Manuel Lopes. Facilitating intention prediction for humans by optimizing robot motions. In *International Conference on Intelligent Robots and Systems (IROS)*, 2015.
22. Freek Stulp, Laura Herlant, Antoine Hoarau, and Genaro Raiola. Simultaneous on-line discovery and improvement of robotic skill options. In *International Conference on Intelligent Robots and Systems (IROS)*, 2014.
23. Freek Stulp and Olivier Sigaud. Policy improvement methods: Between black-box optimization and episodic reinforcement learning. hal-00738463, 2012.

24. Leila Takayama, Doug Dooley, and Wendy Ju. Expressing thought: improving robot readability with animation principles. In *Proceedings of the 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 69–76, 2011.
25. C. Vesper, R.P.R.D. van der Wel, G. Knoblich, and N. Sebanz. Making oneself predictable: reduced temporal variability facilitates joint action coordination. *Experimental brain research*, pages 1–14, 2011.
26. M. Zhao, R. Shome, I. Yochelson, K. Bekris, and E. Kowler. An experimental study for identifying features of legible manipulator paths. In *International Symposium on Experimental Robotics (ISER)*, 2014.

Baptiste Busch received a Master’s Degree in Engineering in Computer Science and applied Mathematics from École Internationale des Sciences du Traitement de l’Information (EISTI), France and a double M.Sc. in Robotics from Warsaw University, Poland and Ecole Centrale de Nantes, France, as part of the European Master on Advanced RObotics (EMARO) program in 2014. He is currently a Ph.D. student at INRIA Bordeaux in the FLOWERS team laboratory. His researches focus on human-robot interaction and using ergonomic methods to improve the workers’ comfort during the interaction.

Jonathan Grizou received a Ph.D. degree in computer science from INRIA, France, in 2014. He investigated calibration-free interactive systems with application to brain-computer interfaces, for which he received “le Prix Le Monde de la Recherche Universitaire 2015”. He is currently a Research Associate in the Cronin Group, School of Chemistry, Glasgow University, UK. His current interest lies at the intersection of chemistry, robotics, and AI, using robots and algorithms as tools to study properties of complex chemical systems.

Manuel Lopes studied electrical engineering and control theory and did a Ph.D. in social learning for robots from Instituto Superior Tecnico. He was a Lecturer at the University of Plymouth before being a Researcher at Inria, France, and currently an associate professor at Instituto Superior Tecnico, Universidade de Lisboa, Portugal. In the past he has made numerous contributions in the field of robotics and artificial intelligence where he studies and develops new learning algorithms. Since 2014 he is coordinating the European Project 3rdHand and has participated in several other national and international projects in robotics, education and computational neuroscience. His current interests include the study of the fundamental mechanisms of learning in machines and animals, and its application to understanding exploration in animals, human-robot collaboration, autonomous robots and intelligent tutoring systems.

Freerk Stulp received his doctorate degree in Computer Science from the Technische Universität München in 2007. He

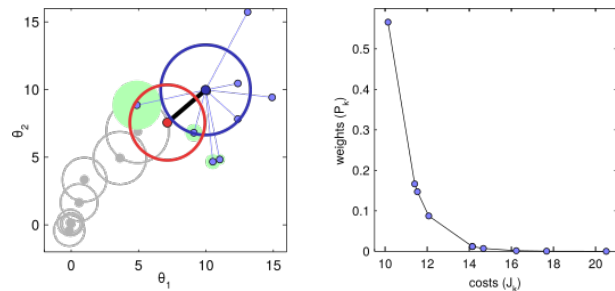


Fig. 12 Illustration of the PI^{BB} algorithm on a simple cost function $J(\theta) = \|\theta\|$ (without policies). Left: iterative updating of the parameters, where the exploratory samples for the first iteration are shown. Right: mapping the costs J_k to weights P_k for the first iteration. The algorithmic parameters are $K = 10$, $\lambda = 0.7$.

was then awarded post-doctoral research fellowships to pursue his research at the Advanced Telecommunications Research Institute International (Kyoto) and the University of Southern California (Los Angeles). After being an assistant professor at the École Nationale Supérieure de Techniques Avancées (ENSTA-ParisTech) in Paris, he is now the head of the department of Cognitive Robotics at the Institute of Robotics and Mechatronics with the German Aerospace Center (DLR), Wessling, Germany. He is an associated member of the FLOWERS team at INRIA Bordeaux. His research interests include robotics, motion primitives, continual robotic learning and semantic planning, with a focus on applications in robust manipulation and future manufacturing.

A Policy Improvement through Black-Box Optimization

Policy improvement is a form of model-free reinforcement learning, where the parameters θ of a parameterized policy π_θ are optimized through trial-and-error interaction with the environment. The optimization algorithm we use is PI^{BB} , short for “Policy Improvement through Black-Box optimization” [23]. It optimizes the parameters θ with a two-step iterative procedure. The first step is to locally *explore* the policy parameter space by sampling K parameter vectors θ_k from the Gaussian distribution $\mathcal{N}(\theta, \Sigma)$, to execute the policy with each θ_k , and to determine the cost J_k of each execution. This exploration step is visualized in Fig. 12, where $\mathcal{N}(\theta, \Sigma)$ is represented as the large (blue) circle, and the samples $J_{k=1\dots 10}$ are small (blue) dots.

The second step is to *update* the policy parameters θ . Here, the costs J_k are converted into weights P_k with

$$P_k = e^{\left(\frac{-h(J_k - \min(J))}{\max(J) - \min(J)}\right)} \quad (6)$$

where low-cost samples thus have higher weights. For the samples in Fig. 12, this mapping is visualized (to the right). The weights are also represented in the left figure as filled (green) circles, where a larger circle implies a higher weights. The parameters θ are then updated with reward-weighted av-

eraging

$$\theta \leftarrow \sum_{k=1}^K P_k \theta_k \quad (7)$$

Furthermore, exploration is decreased after each iteration $\Sigma \leftarrow \lambda \Sigma$ with a decay factor $0 < \lambda \leq 1$. The updated policy and exploration parameters (red circle in Fig. 12) are then used for the next exploration/update step in the iteration.

In the optimization experiments described in this article, the parameters of PI^{BB} are $K = 8$ (trials per update), $\Sigma = 5\mathbf{I}$ (initial exploration magnitude) and $\lambda = 0.9$ (exploration decay).

Despite its simplicity, PI^{BB} is able to learn robot skills efficiently and robustly [22]. Alternatively, algorithms such as PI^2 , PoWER, NES, PGPE, or CMA-ES could be used, see [23, 11] for an overview and comparisons.

B Complete results for Experiment A and B

See next page.

C Complete results for Experiment C

See next page.